Dolev-Yao is no better than Machiavelli*

Paul Syverson, Catherine Meadows

Center for High Assurance Computer Systems
Naval Research Laboratory
Washington, DC 20375-5320, USA
{*syverson, meadows*}*@itd.nrl.navy.mil*

Iliano Cervesato

Advanced Engineering and Sciences Division
ITT Industries, Inc.
Alexandria, VA 22303-1410, USA
iliano@itd.nrl.navy.mil

Abstract

We show that all attacks that can be mounted by a traditional Dolev-Yao intruder against common cryptographic protocols can be enacted by an apparently weaker 'Machiavellian' adversary in which compromised principals will not share long-term secrets and will not send arbitrary messages. We also show that a Dolev-Yao adversary composed of multiple compromised principals is attack-equivalent to an adversary consisting of a single dishonest principal who is only willing to produce messages in valid protocol form.

1 Introduction

Cryptographic protocol analysis traditionally assumes a worst-case scenario. All communication between honest principals passes through a single adversary. Further, the intruder can alter messages in any way within its computational ability as well as change their destination (including blocking them entirely). Worst of all, any compromised principal shares all of his/her information and capabilities with the adversary. For this reason, Anderson and

*The first and second authors were supported by ONR and NSA. The third author is supporting the Formal Methods Section of the Naval Research Laboratory under contract N0014-96-D2024.

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2000		**00-00-2000 to 00-00-2000**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Dolev-Yao is no better than Machiavelli	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Research Laboratory,Center for High Assurance Computer Systems,4555 Overlook Avenue, SW,Washington,DC,20375**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**		**6**	

Needham have described cryptographic protocol design as "programming Satan's computer" [1]. However, this model may be overly pessimistic.

Proposed approaches to weakening the intruder model have been primarily topological, considering a distributed adversary with limited abilities [5, 6, 7]. A complementary possibility is to limit not what the different parts of the adversary *can* do, but what they *are willing* to do. The intruder will perhaps have complete access to signature keys, etc. for a principal that has been overtaken, *e.g.* on a machine for which the adversary has gained root access. But, compromised principals that are not overtaken, but simply dishonest, may be unwilling to share signature keys and other long-term secrets even if they are willing to participate in attacks. We call an adversary composed of such self-interested collaborators '*Machiavellian*' in distinction to the classic Dolev-Yao intruder [4] mentioned above.

It might seem that the adversary composed of Machiavellian collaborators would be less able to mount attacks than a (collection of) Dolev-Yao intruder(s). This work shows that this is not the case for common authentication protocols (that do not transmit long-term secrets). Indeed, not only is a Machiavellian adversary as strong as a Dolev-Yao intruder, but also, surprisingly, all attacks representable with a full blown Dolev-Yao adversary involving multiple compromised principals can be represented using just a single dishonest principal operating alone. We call adversaries capable of mounting the same attacks (in the weakest sense of the term) *attack-equivalent*.

2 Formal Development

In Figure 1, we express a generalization of the Dolev-Yao model to n intruders using the multiset rewriting formalism presented in [2]. The current state of execution of a protocol $\mathcal{P}$ is represented as a multiset of atomic formulas, and each rule prescribes a transition that replaces the elements on the left-hand side with the components in the right-hand side ("$\cdot$" stands for the empty multiset). Objects of the form $\mathsf{N}(m)$ indicate that the message m has been sent on the public network through which honest principals communicate, while each DY_i, for $i = 1..n$, can be seen as the private workshop where Dolev-Yao intruder number i illicitly dismantles and assembles messages. The other predicates (here KeyP and π) hold publicly available information. Observe that the two topmost rules enable the intruders to share all the information they know.

$$
\begin{array}{rcll}
\mathsf{N}(m) & \longrightarrow & \mathsf{DY}_i(m) & (\textit{Interception}) \\
\mathsf{DY}_i(m) & \longrightarrow & \mathsf{N}(m) & (\textit{Injection}) \\
\mathsf{DY}_i(m_1, m_2) & \longrightarrow & \mathsf{DY}_i(m_1), \mathsf{DY}_i(m_2) & (\textit{Decomposition}) \\
\mathsf{DY}_i(m_1), \mathsf{DY}_i(m_2) & \longrightarrow & \mathsf{DY}_i(m_1, m_2) & (\textit{Composition}) \\
\mathsf{DY}_i(\{m\}_k), \mathsf{DY}_i(k'), \mathsf{KeyP}(k, k') & \longrightarrow & \mathsf{DY}_i(m), \mathsf{KeyP}(k, k') & (\textit{Decryption}) \\
\mathsf{DY}_i(m), \mathsf{DY}_i(k) & \longrightarrow & \mathsf{DY}_i(\{m\}_k) & (\textit{Encryption}) \\
\cdot & \longrightarrow & \exists n.\, \mathsf{DY}_i(n) & (\textit{Nonce creation}) \\
\pi(m) & \longrightarrow & \mathsf{DY}_i(m), \pi(m) & (\textit{Public knowledge}) \\
\mathsf{DY}_i(m) & \longrightarrow & \mathsf{DY}_i(m), \mathsf{DY}_i(m) & (\textit{Duplication}) \\
\mathsf{DY}_i(m) & \longrightarrow & \cdot & (\textit{Deletion})
\end{array}
$$

Figure 1: Dolev-Yao Intruder Model

Figure 2 formalizes the generalization to our Machiavellian model to n intruders as a collection of multiset rewrite rules [1]. It differs from the specification of the Dolev-Yao adversary by the imposition of a restriction on the messages that an intruder can send on the network: they shall look like legitimate messages of the protocol. We formalize this idea through the notion of the *skeleton* of a message m, written $sk(m)$, and defined as follows:

$$
\begin{cases}
sk(n) & = \underline{nonce} \\
sk(k) & = \underline{stKey} \\
sk(k') & = \underline{ltKey} \\
sk(m_1, m_2) & = (sk(m_1), sk(m_2)) \\
sk(\{m\}_k) & = \{sk(m)\}_{sk(k)}
\end{cases}
$$

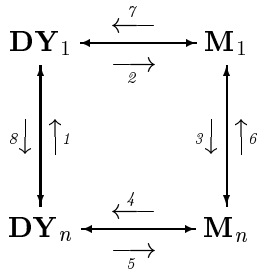
We assume that protocol principals can distinguish short-term secrets (tag $\underline{stKey}$) from long-term keys(tag $\underline{ltKey}$). Indeed, in the following, we shall restrict ourselves to protocols that do not transmit long-term keys, not even encrypted. We also assume that principals know the entire skeleton of any message they receive. The implications of this assumption are further discussed in Section 3. The skeleton of a protocol $\mathcal{P}$, written $sk(\mathcal{P})$, is given by the set of the skeletons of all the messages that are either exchanged as part of the execution of $\mathcal{P}$ or implied by it (e.g., the key built during a Diffie-Hellman exchange).

Our result is summarized in the following diagram, where $\mathbf{DY}_n$ and $\mathbf{M}_n$ stand for the model consisting of n Dolev-Yao and Machiavellian adversaries ($n > 0$), respectively. An arrow from $\mathbf{A}$ to $\mathbf{B}$ indicates that every message

$$
\begin{array}{rcll}
\mathsf{N}(m) & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(m) & (\mathit{Int.}) \\
\mathsf{M}_i(m) & \longrightarrow_{\mathcal{P}} & \mathsf{N}(m) \quad \mathit{if}\ sk(m) \in sk(\mathcal{P}) & (\mathit{Inj.}) \\
\mathsf{M}_i(m_1, m_2) & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(m_1), \mathsf{M}_i(m_2) & (\mathit{Dec.}) \\
\mathsf{M}_i(m_1), \mathsf{M}_i(m_2) & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(m_1, m_2) & (\mathit{Cmp.}) \\
\mathsf{M}_i(\{m\}_k), \mathsf{M}_i(k'), \mathsf{KeyP}(k, k') & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(m), \mathsf{KeyP}(k, k') & (\mathit{Decr.}) \\
\mathsf{M}_i(m), \mathsf{M}_i(k) & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(\{m\}_k) & (\mathit{Encr.}) \\
\cdot & \longrightarrow_{\mathcal{P}} & \exists n.\, \mathsf{M}_i(n) & (\mathit{Nnc}) \\
\pi(m) & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(m), \pi(m) & (\mathit{Pub.}) \\
\mathsf{M}_i(m) & \longrightarrow_{\mathcal{P}} & \mathsf{M}_i(m), \mathsf{M}_i(m) & (\mathit{Dup.}) \\
\mathsf{M}_i(m) & \longrightarrow_{\mathcal{P}} & \cdot & (\mathit{Del.})
\end{array}
$$

Figure 2: Machiavellian Intruder Model

that intruder model **A** can produce, and that may be accepted by an honest principal, can be constructed by adversary model **B**. Therefore, a double arrow between **A** and **B** means that they are attack-equivalent.



The proof of our result proceeds as follows, where the numbering refers to the one-sided arrows in figure.

1 : We reduce n Dolev-Yao adversaries to just one by merging their knowledge and initial data. We achieve this by replacing each piece of state $\mathsf{DY}_i(m)$, for $i = 1..n$, with $\mathsf{DY}(m)$, which will stand for the knowledge of our single target intruder.

2 : We map a single Dolev-Yao adversary to a Machiavellian intruder by observing that the only messages that an honest principal will accept must have a skeleton that conforms to the protocol. Therefore, the only participant who can make use of an intruder-generated message with an unexpected skeleton is the intruder itself. Clearly these trivial

transmission/reception loops can be eliminated. Notice that we need here the ability of a principal to distinguish short-term secrets from long-term keys (and drop messages mentioning the latter).

3, *8* : We simply take n to be 1.

4, *7* : Since the Machiavellian adversary is a restriction of the Dolev-Yao intruder, every message that the former can generate can be produced by the latter.

5, *6* : By transitivity.

We expect to be able to formalize this proof by representing it, for example, in the linear logical framework *LLF* [3].

3 Conclusions and Future Work

The attack equivalence results in this abstract may have implications as far as protocol analysis is concerned. Indeed, different analysis tools may perform more efficiently by using one intruder model rather than another. For example, almost all proposed systems, especially those based on model checking, already assume a single intruder.

Establishing the equivalence of intruder models is non-trivial and can lead to substantial benefits in specific tools. The technique presented here is general, formally based on multiset rewriting concepts [2], and machine-checkable [3]. We intend to use this approach to explore other restrictions to the abilities of the adversary.

One of the factors that contributes to the simplicity of our proofs is the assumption that principals can always establish the skeleton of any message they accept (and produce). This means that a protocol participant knows the type structure of any received message, including any encrypted messages for which that principal lacks the decryption key. It is reasonable to assume that principals can recognize encrypted messages as such (we abstractly reduce signatures to private key encryptions and render hashes as encryptions for which no one has the decryption key). But, it is unrealistic to assume that principals will know the type structure of the submessages contained in such an encryption, unless s/he knows the key. It appears that the assumption can be removed if we make the notions of skeleton and attack-equivalence more subtle. Essentially, attack equivalence must be stated modulo the (sub)messages for which principals do not know the type structure. This also implies a relativization of skeletons to principals and/or

roles. We intend to set out these subtleties and also to further flesh out and explore attack-equivalence in future work.

References

[1] Ross Anderson and Roger Needham. Programming Satan's computer. In J. van Leeuwen, editor, *Computer Science Today*, pages 426–440. Springer-Verlag LNCS 1000, 1995.

[2] Iliano Cervesato, Nancy A. Durgin, Patrick D. Lincoln, John C. Mitchell, and Andre Scedrov. A meta-notation for protocol analysis. In P. Syverson, editor, *Proceedings of the 12th IEEE Computer Security Foundations Workshop — CSFW'99*, pages 55–69, Mordano, Italy, June 1999. IEEE Computer Society Press.

[3] Iliano Cervesato and Frank Pfenning. A linear logical framework. In E. Clarke, editor, *Proceedings of the Eleventh Annual Symposium on Logic in Computer Science — LICS'96*, pages 264–275, New Brunswick, NJ, 27–30 July 1996. IEEE Computer Society Press.

[4] Danny Dolev and Andrew C. Yao. On the security of public-key protocols. *IEEE Transactions on Information Theory*, 2(29):198–208, 1983.

[5] Catherine Meadows. Formal framework and evaluation method for denial of service. In *Proceedings of the 12th Computer Security Foundations Workshop*, pages 4–13, Mordano, Italy, June 1999. IEEE Computer Society Press.

[6] Paul Syverson and Stuart Stubblebine. Group principals and the formalization of anonymity. In J.M. Wing, J. Woodcock, and J. Davies, editors, *FM'99 – Formal Methods, Vol. I*, pages 814–833. Springer-Verlag, LNCS 1708, 1999.

[7] Paul F. Syverson. A different look at secure distributed computation. In *Tenth IEEE Computer Security Foundations Workshop — CSFW-10*, pages 109–115. IEEE Computer Society Press, June 1997.